Journal of KIIT. Vol. 19, No. 6, pp. 109-124, Jun. 30, 2021. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2021.19.6.109

권한 기반 ERBAC

백경동*, 박동규**

Permission based ERBAC

Kyeong-Dong Baek*, Dong-Gue Park**

이 논문은 순천향대학교 연구비에 의하여 연구하였음.

요 약

역할 기반 접근 제어 모델에서 자원에 대한 접근은 사용자가 조직 내에서 보유한 역할을 기반으로 한다. 그러나 역할 기반 접근 제어에서 사용자는 미리 결정된 역할 및 권한에 접근할 수 있다. 따라서 사용자가 때때로 정상적인 상황에서 승인되지 않은 리소스에 접근해야 하는 비상 상황에서 RBAC(Role-Based Access Control) 정책은 원하는 결과를 얻을 수 없다. 본 논문에서는 비상 상황에서 시스템을 관리 시에 RBAC의 유연성을 강화하기 위하여 권한 기반 ERBAC(Emergency RBAC)를 제안한다. 제안된 모델은 비상 상황에서 최소 권한 원칙을 유지하기 위하여 권한 기반 할당 메커니즘을 사용하며, 사용자 접근을 제어하기 위해 BTG-SSD, BTG-DSD 및 BTG-Binding 제약을 포함한다. 그리고 제안된 모델은 대규모 ERBAC 시스템을 관리하고 관리자의 과도한 부담을 줄이기 위하여 관리 모델을 사용한다. 이 접근 방식은 또한 매우 상세한 기관 특성을 고려한 정책을 가능하게 하여 BTG를 더욱 유연하게 하며, 감사기록을 자동화하게 한다. 마지막으로, 모델 사양의 유효성을 분석하기 위하여 권한 기반 ERBAC을 의료 시나리오로 설명한다.

Abstract

In the Role-Based Access Control(RBAC) model access to resources is based on the role the user holds within the organization. But, the users in RBAC gain access to predetermined roles and permissions. Thus, the policy in RBAC does not achieve desired results in emergency situations that users should sometimes gain access to resources not authorized in normal situations. In this paper, we propose an Permission-based Emergency RBAC(ERBAC) to enhance the flexibility of RBAC for managing the system in emergency situation. The proposed model uses a permission-based authorization mechanism to maintain the principle of least-privilege. and includes BTG-SSD, BTG-DSD and BTG-Binding constraint to control user access in emergency situation. Then, it uses an administrative model to manage large ERBAC systems and alleviate administrator's burden. This approach also makes BTG more flexible, allowing for fine-grained facility-specific policies, and even automates auditing in many situations. Finally, it is illustrated with medical scenarios so as to analyze the validity of proposed model specifications.

Keywords
RBAC, emergency, BTG, least-privilege

* 순천향대학교 정보통신공학과 석사과정
- ORCID: https://orcid.org/0000-0002-3918-5511
** 순천향대학교 정보통신공학과 교수(교신저자)
- ORCID: http://orcid.org/0000-0002-5864-8825

· Received: May 07, 2021, Revised: Jun. 18, 2021, Accepted: Jun. 21, 2021
· Corresponding Author: Dong-Gue Park
Dept. of Information and Communication Engineering Soonchunhyang Univ.
Tel.: +82-41-530-1347, Email: dgpark@sch.ac.kr.

Ⅰ. 서 론

많은 정보가 컴퓨터 시스템에 저장이 되고 이러한 정보 중 개인정보 및 기업의 기밀문서와 같은 민감한 정보들도 있기 때문에 무단 접근으로부터 보호되어야 한다. 역할 기반 접근 제어(RBAC, Role Based Access Control)는 컴퓨터 시스템 보안에서 권한이 있는 사용자들에게 시스템의 접근을 통제하는 방법이다[1][2]. 대부분의 조직 및 기관에서는 작업에 대한 다양한 역할을 설정한다. RBAC은 미리 작업에 대한 역할을 설정하고 사용자는 역할에 맞는 작업을 수행한다. 하지만 작업에 대한 권한이 사전에 정의되어 있고 규칙이 올바르게 설정되어 있지만, 오류 및 예상하지 못한 비상 상황이 발생할 수 있기 때문에 기존의 RBAC만으로는 비상 상황에서의 접근제어로 적합하지 않게 된다[3]-[6]. 비상 상황을 해결할 때에는 사용자는 인가되지 않은 자원 접근에 대한 권한이 필요하다. 비상 상황에서 RBAC의 유연성을 높이기 위하여 BTG(Break The Glass) 정책을 도입하여 사용자는 제어된 방식으로 비상 상황에서 인가되지 않은 정보에 접근을 할 수 있는 BTG-RBAC이 도입되었다[3]-[6]. BTG-RBAC 정책에서는 사용자에게 비상 상황을 해결할 수 있는 권한을 부여하는 것과 동시에 모든 활동을 기록 검토함으로써 비상 상황 시의 작업에 대한 책임을 생성한다[3]-[6].

본 논문에서는 다양한 비상 상황을 해결하기 위해 사용자의 권한을 기반으로 하는 ERBAC (Emergency RBAC) 모델을 제안한다. 정상 상황 일 시에는 기존의 RBAC 정책에 사용자의 권한을 침해하지 않고 최소 권한 원칙을 고수하기 위하여 권한 기반으로 정적 의무 분리(SSD, Static Separation of Duty), 동적 의무 분리(DSDm Dynamic Separation of Duty)[7][8] 및 바인딩(Binding)을 고려하고, 비상 상황 시에는 사용자가 BTG 권한 요청 시 해당 기관의 특성을 고려한 사용자 신뢰 수준과 BTG 상황에서만 적용되는 권한 기반의 BTG-SOD 및 BTG-Binding의 개념을 새로 도입하여 사용자가 상황을 해결할 수 있도록 관리자가 권한을 부여한다. 그리고 비상 상황 시 행하여야 하는 의무 수행 여부에 따라서 시스템의 상태를 제어 모드, 비 제어 모드 두 가지의 경우로 나누어 시스템이 의무를 수행할 수 있는 상황이면 제어 모드로 비상 상황에서의 작업 기록이 모두 자동으로 감사기록을 수행하도록 하고, 의무를 수행하지 못하는 경우에는 비 제어 모드로 비상 상황이 종료된 후 관리자가 직접 감사기록을 수행하도록 하여 의무를 수행하지 못하는 경우에도 비상 상황 처리가 가능하도록 하게 한다[9]. 또한, 제한된(Restricted) BTG 자원 개념[9]을 도입하여, 비상 상황에서도 특정 자원에는 절대 접근을 할 수 없도록 하여 시스템의 안전성으로 높이도록 접근 제어를 세밀하게 수행한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구, 3장에서는 본 논문에서 제안하는 접근 제어 모델, 4장에서는 제안된 모델의 예시를 설명하고 5장에서는 기존 연구와의 비교평가를 기술하며 마지막으로 6장은 결론으로 구성된다.

Ⅱ. 관련 연구

접근 제어의 개념은 1970년대에 처음 제안되었으며, 이후 임의 접근 제어(DAC, Discretionary Access Control), 의무적 접근 제어(MAC, Mandatory Access Control) 등 여러 접근 제어가 등장하였지만 역할 기반 접근 제어가 가장 많이 사용되었다[2]. RBAC은 NIST(National Institute of Standard and Technology)에서 NIST RBAC 표준을 제안하였으며[2], RBAC 모델에서는 권한과 사용자 사이에 역할을 배치하여 권한 관리에 사용한다. RBAC에서는 최소 권한 개념과 내부 보안 공격 및 위협으로부터 조직을 보호하기 위하여 의무 분리를 사용한다. 의무 분리는 사용자에게 시스템을 오용할 수 있는 충분한 권한을 부여하지 않는 원칙으로[10], 두 가지가 있다. 첫 번째는 정적 의무 분리로 사용자에게 역할을 할당할 때 서로 상충하는 두 가지 권한을 한 사용자에게 할당할 수 없는 것을 말하며, 두 번째는 동적 의무 분리로 사용자에게 서로 상충하는 두 가지의 권한을 부여할 수는 있지만 동시에 활성화가 될 수 없다는 것이다[10]. 또 다른 제약조건으로 바인딩은 사용자가 특정 역할을 수행하면 연이어서 다른 역할도 수행하여야 한다는 것이다[4].

Habib 등[7] 연구자들은 기존 RBAC 모델에서 동

적 의무 분리와 관련된 RBAC 표준의 문제점을 논의하고 역할보다는 객체 기반의 권한 수준에서 동적 의무 분리를 구현하는 모델을 제안했다. 그러나 이 모델도 일반적인 상황에서만 고려하였고 관리자의 부담이 증가하는 단점이 있었다. Habib 등[8] 연구자들은 권한 수준 업무 분리를 고려하여 최소 권한 원칙을 준수하였고 역할 및 권한에 속성을 도입하여 관리자의 부담을 줄였다. 그러나 이 모델도 권한 바인딩 개념을 고려하지 않았고 비상 상황을 고려하지 못한 단점이 있었다.

RBAC은 자가 관리에 적용할 수 있다는 장점이 있지만 사용자, 역할 및 권한이 많은 경우 보안 관리자가 시스템을 관리하는 것이 매우 복잡해진다[5][11]. 이 문제를 해결하기 위하여 ARBAC97[11]에서는 관리 모델을 도입하여 관리자의 부담을 줄였다. ARBAC97은 관리 역할 계층에 의해 역할에 대한 권한 및 사용자에 대한 역할 할당을 제어하며, 각 관리 역할의 역할 범위를 결정한다. ARBAC97에는 URA97(사용자 대 역할 할당), PRA97(권한 대 역할 할당), RRA97(역할 대 역할 할당) 및 역할 범위, 관리 역할 및 일반 역할과 같은 개념이 포함된다.

RBAC에서의 모든 역할에는 결정된 권한이 있으나 실제 세계에서 사용자는 때때로 정상적인 상황에서 승인되지 않은 자원에 접근해야 하는 상황이 발생하게 된다. 이 문제를 해결하고 접근 제어의 유연성을 높이기 위해 위임(Delegation)과 BTG라는 두 가지 메커니즘이 제안되었다[12][13]. 위임에서는 사용자가 자신의 업무를 다른 사용자에게로 전송할 수 있으며, BTG 메커니즘은 사용자가 제어된 방식으로 접근 제어 정책을 재정의 할 수 있도록 한다[12]. 따라서 사용자는 일반적으로 정상적으로는 접근할 수 없는 권한에 비상 상황에서 접근할 수 있게 된다[3][5][6][9]. 하지만 사용자가 무단으로 다른 역할에 접근을 방지하는 것과 사용자에게 책임을 부여하기 위하여 BTG 상황 시 특정 역할에 접근하려면 먼저 의무가 수행되어야 한다. 의무란 예를 들어, 긴급 상황에서 자신에게 없는 역할을 긴급 상황을 해결하기 위해 역할을 요청하면 역할을 할당받기 전 관리자에게 보고하는 것이나 역할을 할당받아 일을 해결하는 동안 사용자가 한 행위에 대한 모든 일을 기록하여 감사에 보고하는 것 등이 있다[3][5][6][9]. Ferreira등[3]은 권한의 BTG 상태를 보유하고 있는 BTG_RBAC 엔진을 탑재한 BTG-RBAC 모델을 제안하였다. BTG-RBAC 모델은 BTG 상태가 TRUE이면 권한을 역할에 할당할 수 있고 BTG-RBAC의 모든 활동은 추후 비상 상황 시의 동작에 대한 적법성을 증명하기 위하여 감사기록을 수행해야 한다. 이 모델은 비상 상황 시의 접근제어에 대한 모델을 제시하였으나, 업무 분리(SOD) 및 업무 바인딩(BOD)과 같은 제약을 고려하지 못한 단점을 가지고 있다. Sigrid 등[4]에서는 프로세스 기반의 RBAC 모델에 BTG 개념으로 도입하여 업무 분리 및 업무 바인딩과 같은 제약을 포함해서 접근제어 결정을 수행하였다. 그러나 이 모델도 기존의 프로세스 처리 시 프로세스 기반으로 비상 상황을 처리하고 있으며, 권한 기반의 비상 상황을 고려하지 않고 있는 단점이 있다. 또한 비상 상황 시에 기관의 특성으로 고려하지 못하고 있으며 관리자의 부담이 커지는 단점이 있다.

Nazerian 등[5]은 RBAC를 비상 상황 시에 적용할 수 있도록 RBAC을 E-RBAC으로 확장했다. E-RBAC에서는 시스템을 정상, 응급 상황 및 예외 상황 등 세 가지 상태 중의 하나를 갖도록 하였다, 정상 상태는 접근제어 무효화가 없는 RBAC과 유사하며, 반면에 응급 상황은 정의되어 있는 비상 상황으로 사용자가 미리 결정된 적절한 신뢰 레벨(응급 상황 M 그리고 예외 상황 H)을 가진 경우에만 접근제어를 무효화할 수 있다. 예외 상황은 정의되지 않은 응급 상황을 나타내며, 사용자의 활성화된 역할에 허가를 할당하거나 취소하게 된다. 그리고 E-RBAC에서는 비상 상황에서의 역할 할당 시 관리자의 부담을 줄이기 위하여 관리모델을 사용하였다. 그러나 E-RBAC에서는 역할을 기반으로 비상 상황을 처리하기 때문에 최소 권한 부여의 원칙이 지켜지지 않으며, 비상 상황 시 사용자에게 역할을 할당하기 위하여 세 가지 상황으로 분류하여 역할을 할당하기 때문에 관리자의 부담이 증가하는 단점이 있었다.

또한, 해당 모델에서는 의무 사항이 수행되지 못하는 경우에 비상 상황의 접근 제어 처리가 안 되는 단점이 있으며, 역할을 기반으로 한 정적 의무

분리만을 고려하여 일반 상황 시의 정적 의무 분리를 그대로 비상 상황 시에 적용함으로써 비상 상황 시의 권한 할당에 대한 제약이 존재한다.

의무 사항이 수행되지 못하는 경우에도 비상 상황 시 접근제어 처리를 위하여 수작업 감사기록이 수반되는 uncontrolled BTG[9] 논문이 제안되었다. 해당 논문[9]은 ABAC기반으로 정상, controlled BTG, uncontrolled BTG로 시스템의 상태를 분류하여 의무 사항이 수행되지 못하는 경우에도 관리자가 수작업으로 감사기록을 수행함으로써 비상 상황의 권한 할당을 가능하게 하였다. 그러나 해당 논문도 접근제어를 수행하는 기관의 고유 특성을 고려하지 못한 단점이 있으며, BTG 상태에서의 업무 분리와 바인딩을 고려하지 못한 단점이 있었다.

본 논문에서는 이런 문제점들을 해결하고 다양한 비상 상황에서 유연한 접근제어를 수행하기 위하여 사용자의 권한을 기반으로 하는 권한 기반 ERBAC 모델을 제안한다. 권한 기반 ERBAC모델에서 정상 상황일 시에는 기존의 RBAC 정책과 동일하게 작동하며, 권한을 기반으로 한 정적 의무 분리, 동적 의무 분리 및 권한 기반 바인딩을 고려하고, 비상 상황 시에는 사용자가 BTG 권한 요청 시 행하여야 하는 의무 수행, 신뢰 수준과 BTG 상황에서만 적용되는 권한 기반의 BTG-SSD, BTG-DSD 및 BTG -Binding의 개념을 새로 도입하여 사용자가 상황을 해결할 수 있도록 한다. 또한, 비상 상황 시에 시스템의 상태를 제어 모드, 비 제어 모드 두 가지의 경우로 나누어 시스템이 의무를 수행할 수 없는 경우에도 효율적으로 비상 상황을 처리할 수 있도록 한다. 또한, 제한된 BTG 자원 개념[9]을 도입하여, 비상 상황에서도 제한된 BTG 자원에는 절대 접근을 할 수 없도록 하여 시스템의 안전성으로 높이도록 접근 제어를 세밀하게 수행한다.

III. 권한 기반 ERBAC

3.1 권한 기반 ERBAC 모델

본 논문에서 제안하는 권한 기반 ERBAC 모델은 그림 1과 같다.

제안된 권한 기반 ERBAC은 USERS(사용자 집합), ROLES(역할 집합), AR(관리 역할 집합), OPS (작업 집합), OBS(객체 집합), SESSIONS(세션 집합), OBLGS(권한을 얻기 전에 수행 하여야 하는 의무 집합), AP(관리 권한 집합), Au(사용자 속성 집합)등의 기본 요소가 포함된다. CONSTRAINTS(제약조건)은 권한 기반 정적 의무 분리와 권한 기반 동적 의무 분리 및 권한 기반 바인딩의 조건을 나타내며, BTG-SSD와 BTG-DSD 및 BTG-Binding 조건을 나타낸다. 그리고 제한된 BTG 자원을 표시하며, 또한 정책을 입안하는 기관 및 조직에 따라 설정 가능한 사용자의 속성을 나타낸다.

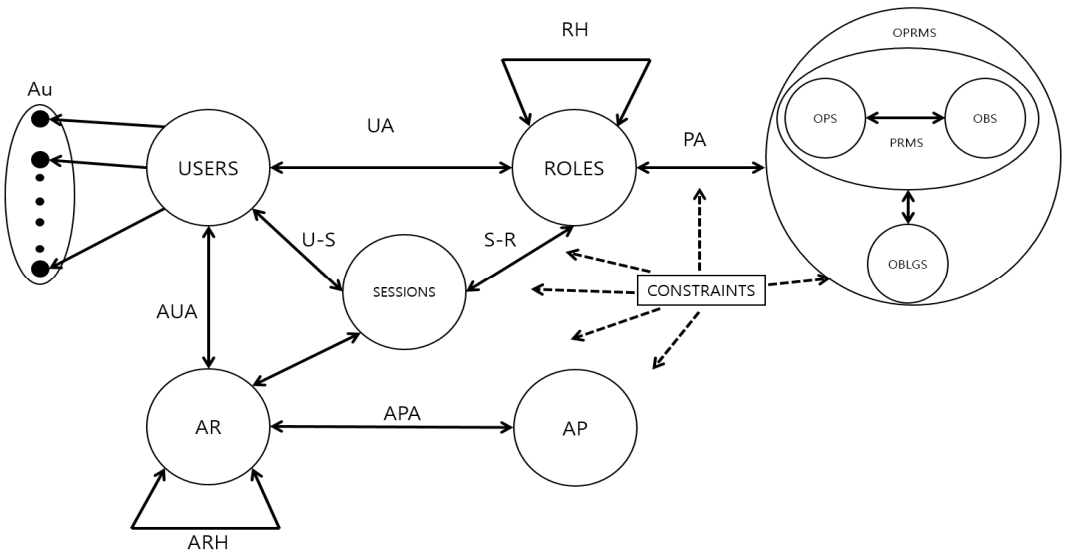

그림 1. 권한 기반 ERBAC 모델
Fig. 1. Permission-based ERBAC model

정의 1. 권한 기반 ERBAC

권한 기반 ERBAC의 관계는 다음과 같다.

- U(USERS): 사용자 집합
- UA ⊆ USERS×ROLES는 사용자와 역할의 관계
- R(ROLES): 역할 집합
- S(SESSION): 세션 집합
- U-S(U: USERS)→ $2^{SESSION}$은 사용자와 세션의 관계
- S-R(S: SESSION)→ 2^{ROLES}는 세션과 역할의 관계
- PRMS = OPS×OBS는 객체에 대한 작업을 수행하는 권한 집합
- OPRMS ⊆ PRMS×2^{OBLGS}는 권한과 의무의 관계
- PA ⊆ OPRMS×ROLES는 역할과 권한의 관계
- RH ⊆ R×R은 역할 계층 관계
- AR(Administrator Role) : 관리자 역할 집합
- AP(Administrator Permission) : 관리자 권한 집합
- AUA ⊆ U×AR은 사용자와 관리자의 역할 관계
- ARH ⊆ AR×AR은 관리 역할 계층관계
- CONSTRAINTS: 제한된 BTG 자원, 권한 기반 정적 의무 분리, 권한 기반 동적 의무 분리, 권한 기반 바인딩, 권한 기반 BTG-SOD 및 BTG-Binding의 제약 조건
- 권한 소유권 매핑(Pwon) R(ROLES) -> 2OPRMS : 직접 또는 역할 계층 전이를 통해 역할 r에 할당된 모든 권한 집합
- 매핑 pown-1: OPRMS → 2Roles: 직접 또는 역할 계층 전이를 통해 특정 권한을 소유하고 있는 모든 역할 집합
- 매핑 p_binding: oprms → 2oprms는 권한 기반 바인딩 (p_binding (oprms1) = OPRMSp_binding의 경우 oprms1을 권한 기반 바인딩 된 권한이라고 하고 OPRMSp_binding ⊆ OPRMS를 권한 기반 바인딩 된 권한 집합이라고 한다.)
- 매핑 p_ssd: oprms → 2oprms 를 권한 기반 정적 의무 분리 (p_ssd(oprms1) = OPRMSp_ssd with OPRMSp_ssd ⊆ OPRMS의 경우에 각 쌍 oprms1 및 oprmsx ∈ OPRMSp_ssd을 권한 기반 정적 의무 분리된 권한 집합이라고 한다.)
- 매핑 p_dsd: oprms → 2oprms 를 권한 기반 동적 의무 분리 (p_dsd(oprms1) = OPRMSp_dsd with OPRMSp_dsd ⊆ OPRMS의 경우에 각 쌍 oprms1 및 oprmsx ⊆ OPRMSp_dsd 을 권한 기반 동적 의무 분리된 권한 집합이라고 한다.)
- 매핑 rown: U → 2Role 사용자가 직접 또는 역할 계층 전이를 통해 상속 되어 소유하고 있는 모든 역할 집합
- 매핑 $rown^{-1}$: R(Role) → 2S는 특정 역할을 직접 및 역할 계층 전이를 통해 상속 되어 소유하고 있는 모든 사용자 집합
- 매핑 active_session (u): 사용자 u가 활성화된 세션을 통해 소유하고 있는 역할 집합
- 매핑 active_session-1(r): 역할 r을 활성화된 세션을 통해 소유하고 있는 사용자 집합

본 논문에서 제안하는 권한 기반 ERBAC은 정상 상황과 비상 상황 두 가지의 범주로 분류하여 정의한다. 정상 상황 시에 본 논문에서 제안한 모델은 정의 2와 같이 권한 기반 업무 분리와 권한 기반 바인딩을 적용한다. 권한 기반 정적 의무 분리와 동적 의무 분리란 권한을 기반으로 정적 의무 분리와 동적 의무 분리를 제한하기 때문에 사용자는 하나 이상의 충돌되는 역할에 접근할 수 있지만, 서로 상충하는 두 가지 권한에만 접근할 수 없기 때문에 사용자가 할당받은 역할에서 사용 가능한 권한이 역할 기반 의무 분리로 인하여 사용 못 하게 되는 제약이 없어지게 된다. 따라서 사용자의 권리를 침해하지 않고, 최소 권한의 원칙을 유지할 수 있게 된다[7][8]. 이 외의 동작은 표준 RBAC[2]와 같이 동작한다. 권한 기반 ERBAC 모델은 정상 상황 시에 정의 2와 같은 요구 사항이 충족되는 경우 동적으로 정확하다고 한다.

정의 2. 정상 상황 동적 정확성

1. (권한 기반 정적 의무 분리 관계: p_ssd) 각 사용자는 정상적으로 역할 구성원을 통해 pown(r)을 통해 권한을 소유 할 수 있다. 정상 권한이 p_ssd 관계에 있다면, 권한 소유권 매핑을 통해 특정 사용자에 p_ssd 관계에 있는 권한이 할당되지 않도록 해야 한다.

$\forall$ oprms1, oprms2 $\in$ OPRMS, oprms1 $\in$ p_ssd(oprms2), r_1, $r_2 \in$ R

oprms1 $\in$ pown(r_1), oprms2 $\in$ pown(r_2), $rown^{-1}$ (r_1) $\cap$ $rown^{-1}$(r_2) = $\varnothing$

2. (권한 기반 동적 의무 분리 관계: p_dsd) 각 사용자는 정상적으로 역할 구성원을 통해 pown(r)을 통해 권한을 소유 할 수 있다. 정상 상황에서 권한이 p_dsd 관계에 있다면, 정상 권한 소유권 매핑을 통해 권한을 가지고 있는 특정 사용자가 p_dsd 관계에 있는 권한들을 동시에 활성화해서는 안된다.

$\forall$ oprms1, oprms2 $\in$ OPRMS, oprms1 $\in$ p_dsd(oprms2), r_1, $r_2 \in$ R
oprms1 $\in$ pown(r_1), oprms2 $\in$ pown(r_2),
active_session^{-1} (r_1) $\cap$ active_session^{-1} (r_2) = $\varnothing$

3. (권한 기반 바인딩 관계: p_binding) 각 사용자는 정상적으로 역할 구성원을 통해 pown(r)을 통해 권한을 소유 할 수 있다. 정상 상황에서 권한이 바인딩 관계에 있다면, 정상 상황에서 권한 소유권 매핑을 통해 특정 사용자에 p_binding 관계에 있는 권한이 할당되어야 한다.

$\forall$ oprms1, oprms2 $\in$ OPRMS, oprms1 $\in$ p_binding(oprms2), r_1, $r_2 \in$ R
oprms1 $\in$ pown(r_1), oprms2 $\in$ pown(r_2),
$rown^{-1}(r_1) = rown^{-1}(r_2)$

비상 상황이 발생하면 본 논문에서 제안한 모델에서 사용자는 시스템에 비상 상황 해결을 위하여 필요한 권한을 요청하고, 시스템은 이 순간부터 모든 작업을 기록한다. 시스템은 권한과 관련된 의무를 수행할 수 있는지, 없는지를 확인한 후 의무를 수행할 수 있으면 제어 모드로 전환하고 의무를 수행할 수 없는 상태이면 비 제어 모드로 전환한다. 그 후 시스템은 사용자 신뢰 수준과 제한된 BTG 자원인지 여부를 점검하고, BTG-SSD, BTG-DSD, BTG-Binding 및 의무 수행 여부 등 여러 제약 조건들을 고려하여 권한을 부여한다. 사용자는 이후에 비상 상황에서 할당받은 BTG 권한을 사용하여 비상 상황을 해결하게 된다. 그 후 사용자는 비상 상황이 해결됐다고 관리자에게 알리면 시스템의 상태가 제어 모드인 경우에는 로그 기록은 감사 기록에 자동으로 저장이 되고, 비 제어 모드인 경우는 로그 기록을 관리자가 수동으로 감사기록에 저장한 후에 시스템의 상태는 정상 상태로 복귀한다.

3.2 사용자 신뢰도

본 논문에서 제안한 권한 기반 ERBAC에서는 비상 상황 시에 요청한 권한의 인가와 관련하여 각 기관의 특성을 반영하기 위하여 사용자의 속성 집합(Au)에 기반 한 사용자 신뢰 수준(Ut) 개념[5]을 사용한다. 신뢰 수준은 각 사용자에 대해 정의되며 사용자 신뢰 수준을 계산하기 위해 사용자 속성 집합(Au)을 정의한다. 이러한 속성들은 권한 기반 ERBAC 정책을 입안하는 조직과 기관의 특성에 따라 조율될 수 있다. 본 논문에서는 사용자 속성 집합의 예로 사용자 근무 태도(Aa), 사용자 직위(Ab) 및 사용자 근무 일수(Ac)등을 사용한다. 사용자의 속성 집합은 해당 기관의 관리자가 결정한 비상 상황 시의 사용자를 신뢰할 수 있는 임계 값 T(Threshold)를 기준으로 'L'(LOW), 'H'(HIGH)로 나누어진다.

사용자의 신뢰 수준(Ut)을 계산하는 예는 다음과 같다.

$$U_t = \frac{(\alpha \times A_a) + (\beta \times A_b) + (\gamma \times A_c)}{A_a + A_b + A_c},$$
$$(0 < \alpha, \beta, \gamma < 1\ )$$

세 가지 매개 변수 α, β, γ는 각각 속성 Aa, Ab 및 Ac의 영향률로 정의되며, 영향률 및 각 속성의 상한은 조직의 관리자에 의하여 정의되어 각 기관의 특성이 반영될 수 있다. 또한 관리자는 Ut에 대한 임계 값을 설정하여, 임계 값을 기준으로 신뢰 수준이 낮은 사용자는 L로 그 보다 높은 사용자는 H 로 결정한다.

사용자 신뢰수준 L, H는 다음과 같다.

Ut = 'L': 신뢰 수준인 T 보다 낮은 사용자로 정상적인 상황에서 활동할 수 있다.

Ut = 'H': 신뢰 수준이 T 보다 높은 사용자는 정상 및 비상 상황에서 활동할 수 있다.

3.3 BTG-SSD, BTG-DSD 및 BTG-Binding

기존의 몇몇 연구[3][5][6]에서는 역할을 기반으로 한 비상 상황을 도입하였지만 본 논문에서는 제안한 권한 기반 ERBAC모델에서는 권한을 기반으로 하는 비상 상황을 도입한다. 기존의 논문들은 비상 상황을 해결하기 위하여 비상 역할을 할당하거나, 필요로 하는 권한을 역할에 할당하여 사용자에게 권한을 사용할 수 있도록 하였다[5][6].

그러나 이 방식을 사용하면 비상 상황을 두 가지 방식을 나누어서 처리해야 함으로써 처리가 복잡해지고 관리자의 부담이 커지는 문제가 발생하게 된다, 또한 역할을 기반으로 의무 분리를 고려하기 때문에 의무 분리의 관계에 있는 역할들 중 상충되지 않은 권한들도 사용자가 사용하지 못하게 되는 문제가 발생하게 된다[7][8].

이를 해결하기 위하여 본 논문에서 제안한 권한 기반 ERBAC모델은 정상 상황에서 권한을 기반으로 하는 권한 기반 의무 분리 및 바인딩 제약 조건을 도입하여 사용자의 사용할 수 있는 권한을 보호하면서 최소 권한 부여의 원칙을 가능하게 한다. 또한, 비상 상황에서는 비상 상황 해결을 위하여 권한을 기반으로 사용자에게 최소 권한만을 할당하여 줌으로써 관리자의 부담도 줄이고 최소 권한 할당의 원칙을 고수할 수 있다. 정상 상황에서의 권한 기반 의무 분리 및 바인딩뿐만 아니라 비상 상황에서도 필요한 의무 분리 및 바인딩 제약조건 접근제어를 유지하기 위하여 비상 상황에서도 준수되어야 하는 권한 기반의 BTG 의무 분리와 BTG 바인딩 개념을 도입하여 시스템의 안전성을 높이도록 한다.

3.4 제어 모드와 비 제어 모드

E-RBAC[5]에서는 비상 상황에 있어 필요 권한을 요청할 경우 의무 수행이 안 되는 경우에 권한 요청이 허가되지 않는 문제가 있다. 반면 controlled BTG[9]에서는 비상 상황에 있어 필요 권한을 요청할 경우 의무 수행을 충족하지 못하는 경우에도 시스템의 상태를 제어 모드(Controlled)와 비 제어 모드(Uncontrolled) 두 가지 경우로 나누어 보다 효율적으로 BTG 상황을 해결할 수 있도록 하였다. 본 논문에서 제안한 권한 기반 ERBAC에서도 의료 시스템에서 비상 상황인 경우에 시스템의 상태를 제어 모드와 비 제어 모드 두 가지 경우로 나누어 효율적으로 비상 상황을 처리한다. 권한 기반 ERBAC에서 비상 상황 시 사용자가 권한을 요청하고 주어진 의무를 모두 수행할 수 있는 상황일 때 시스템의 상태는 제어 모드로 설정되고, 권한을 요청하였으나 만약 주어진 의무 중 하나라도 수행이 불가능한 경우 시스템의 상태는 비 제어 모드로 설정하여, 비상 상황이 해결되었을 때 시스템의 상태가 제어 모드인 경우에는 비상 상황 시의 로그 기록은 감사기록에 자동으로 저장이 되고, 비 제어 모드인 경우는 로그 기록을 관리자가 수동으로 감사기록에 저장한 후에 시스템의 상태를 정상 상태로 복귀함으로써 비상 상황을 효율적으로 해결한다.

3.5 제한된 BTG 자원

응급 상황에서 의료 시스템의 이용성은 다른 보안 요소들 보다 우선시 되어야 하지만 환자에 연결된 장치나, 이 장치들을 제어하는 애플리케이션에 허가되지 않은 접근은 환자를 위해하거나, 프라이버시 침해 또는 사망에 이르게 하는 결과를 초래할 수 있다. 이런 상황을 해결하기 위하여 본 논문에서 제안한 권한 기반 ERBAC은 제한된 BTG 자원 개념을 도입하여 비상 상황 시에도 특정 자원에 대한 접근은 할 수 없도록 제어하여 시스템의 안전성을 보장하도록 한다.

3.6 관리(Administrative) 모델

사용자, 역할 및 권한이 많은 경우 보안 관리자 한명이 관리하기에는 복잡하고 부담이 많이 된다. 이를 해결하기 위한 솔루션으로 관리 모델이 제안되었다. 대규모 RBAC[2]은 ARBAC97[11]과 같은 관리 역할 기반 접근 제어 모델을 통해 관리 된다. ARBAC97[11]은 권한-사용자 할당, 권한-역할 할당 등 권한을 추가 또는 제거를 한다. 본 논문에서는 ARBAC97[11]을 확장하여 비상 상황에서 권한 할당을 제어한다.

정상적인 상황에서는 ARBAC97[11]와 같은 개념

으로 사용된다. ARBAC97에서 URA97은 can_assign() 및 can_revoke()를 사용하여 사용자-역할 할당을 확인한다. 그리고 PRA97은 can_assignp() 및 can_revokep()를 사용하여 권한-역할 할당을 확인한다. 여기에서 can_assignp(ar, pc, rr)는 URA와 마찬가지로 관리 역할(ar)이 전제 조건(pc)를 확인하여 참인 경우 역할(rr)에 권한을 할당할 수 있다. 또한 can_revokep(ar, rr)을 통하여 관리 역할(ar)은 역할(rr)의 권한을 제거할 수 있다. RRA97은 관리 역할(ar)이 캡슐화 된 범위 역할(rr)에서 역할을 추가하거나 제거할 수 있는 관계 can_modify(ar, rr)를 포함하는 역할 계층을 확인한다.

본 논문에서 제안한 권한기반 ERBAC 모델에서는 비상 상황 시 사용자가 요청하는 비상 상황 권한을 사용자에게 할당해 주면 된다. 따라서 PRA97만 사용하면 되고, 권한을 역할에 할당 하는 경우는 can_assignp(ar, pc, r)에 의해 제어 될 수 있다. 여기서 ar은 관리 역할, r은 권한을 요청하는 사용자 역할, pc는 전제 조건이다. 전제 조건에서는 사용자의 신뢰 수준, BTG-SOD(BTG-SSD와 BTG-DSD) 및 BTG-Binding제약 조건이 확인된다. 권한 취소는 can_revokep(ar, r)에 의해 제어 될수 있다. 권한 할당 순서는 다음과 같다. 사용자 ‘u'가 권한 ’p'를 요청하고 관리 역할 ‘ar’은 관계 can_assignp(ar, pc, r)을 사용한다고 가정한다. 여기서 전제 조건 pc는 다음과 같다.

(label(u : USERS) = H) ∧ (p ∉ restricted-BTG-resource) ∧ (∀(ps, n) ∈ BTG-SSD • (pown(rown(u)) ∪ p) ∩ BTG-SSD(ps, n)) = ∅) (∀(ps, n) ∈ BTG-DSD • ((pown(rown(u)) ∪ p) ∩ BTG-DSD(ps, n)) = ∅) ∧ ((∀(ps, n) ∈ BTG-Binding • (p ∩ BTG-Binding(ps, n) ≠ ∅)) -> p = BTG-Binding(p))

전제 조건에서 label(u : USERS) = H)은 사용자 신뢰 수준을 확인하고, (p ∉ restricted-BTG-resource)는 요청하는 권한이 restricted-BTG-resource에 속하지 않아야 한다. 그리고 rown(u)는 사용자 'u'가 소유하고 있는 역할을 반환하고, pown(r)는 역할 r이 소유한 권한 ‘p’를 반환한다. 그래서 pown(rown(u))는 사용자 'u'가 소유하고 있는 역할을 통해서 소유하고 있는 권한 ‘p’를 반환하게 된다. 그 후 BTG-SSD와 BTG-DSD 조건에 위배 되지 않는 지를 확인한다. 그 이후에 BTG-Binding의 제약 조건을 확인하여 요청하는 권한 p가 BTG-Binding의 제약 조건에 해당하면 권한 p가 바인딩 된 권한들(BTG- Binding(p))을 포함하여 사용자에게 할당하고, 그렇지 않으면 요청하는 권한 p를 사용자에게 할당된다. 여기에서 BTG-Binding(p)는 권한 p가 속한 BTG-Binding 권한 집합을 반환한다. 비상 상황이 종료되면 관리자는 can_revokep(ar, r)를 사용하여 비상 상황에서 사용자에게 할당한 권한을 회수하게 된다.

3.7 권한 기반 ERBAC 비상 상황 정책 모델

권한 기반 ERBAC 모델에서는 다음 정의 3 ~ 정의 6과 같이 비상 상황 정책에 대한 매핑을 정의한다.

정의 3. 비상 상황 권한 할당 및 매핑

권한 기반 ERBAC 모델은 비상 상황 시에 다음과 같이 관리자 역할을 사용하여 비상 상황 권한을 역할에 할당 및 취소한다.

1. 비상 상황 시에 다음과 같이 관리자 역할을 사용하여 can_assignp를 통하여 비상 상황 권한을 사용자의 역할에 할당 한다. 사용자는 역할에 할당된 비상 상황 권한을 비상 상황에서 실행할 수 있다.
   매핑 bbr : R(ROLES) -> 2OPRMS can_assignp를 통하여 사용자의 역할에 할당된 비상 상황 권한 bbr(r) = OPRMSb의 경우에 우리는 r ∈ R 역할이라고 하고 OPRMSb ⊆ OPRMS는 r에 할당 된 비상 상황 권한 집합이라고 한다, 매핑 bbr^{-1} : OPRMS → 2Roles는 bbr 매핑을 통해 특정 권한이 할당 된 모든 역할을 반환한다.
2. bbr 매핑은 특정 역할에 할당 된 모든 비상 상황 권한을 결정하는 매핑을 의미한다. 역할 계층 구조에서 각 역할은 이 역할에 직접 할당 된 권한과 하위 역할에서 상속된 권한을 소유한다.
   매핑 bpown : R(ROLES) -> 2OPRMS를 비상 상황에서 역할이 소유하고 있는 권한 집합들이다.

각 r ∈ R에 대해 비상 상황에서 이 역할에 직접 할당 된 권한과 하위 역할에서 상속 된 권한이 포함 된다.

매핑 $bpown^{-1}$: OPRMS → 2Roles는 비상 상황에서 (직접 또는 역할 계층 전이)을 통해 권한이 할당되는 모든 역할 집합을 결정한다. bpown 매핑은 권한 소유권 매핑(pown)을 보완한다.

정의 4. 비상 상황 의무 분리 및 바인딩(binding)

권한 기반 ERBAC 모델은 비상 상황 시에 다음과 같은 BTG-SSD,BTG-DSD 및 BTG-Binding을 정의한다.

1. 매핑 btg_ssd : oprms → 2oprms를 비상 상황 권한 정적 의무 분리라고 한다.
   btg_ssd(oprms1) = OPRMSbtg_ssd with OPRMSbtg_ssd ⊆ OPRMS의 경우에 각 쌍 oprms1 및 oprmsx ∈ OPRMSbtg_ssd을 비상 상황 권한 정적 의무 분리 권한이라고 한다.
2. 매핑 btg_dsd : oprms → 2oprms를 비상 상황 권한 동적 의무 분리라고 한다.
   btg_dsd(oprms1) = OPRMSbtg_dsd with OPRMSbtg_dsd ⊆ OPRMS의 경우에 각 쌍 oprms1 및 oprmsx ∈ OPRMSbtg_dsd을 비상 상황 권한 동적 의무 분리 권한이라고 한다.
3. 매핑 btg_binding : oprms → 2oprms를 비상 상황 권한 바인딩이라고 한다.
   btg_binding (oprms1) = OPRMSbtg_binding의 경우 oprms1을 비상 상황 권한 binding된 권한이라고 하고 OPRMSbtg_binding ⊆ OPRMS를 비상 상황 권한 바인딩 된 권한 집합이라고 한다.

정의 5. 비상 상황 정적 정확성

권한 기반 ERBAC 모델은 다음 요구 사항이 충족되는 경우 정적으로 정확하다고 한다.

1. 각 역할은 정상적으로 또는 비상 상황 시 할당을 통해 권한을 소유 할 수 있다. 정상 권한 소유권과 비상 상황 권한 소유권을 분리하려면, 두 매핑을 통해 하나의 특정 역할에 같은 권한이 할당되지 않도록 해야 한다.

   ∀ oprms ∈ OPRMS : $pown^{-1}$ (oprms) ∩ $bpown^{-1}$ (oprms) = ∅

2. 각 사용자는 정상적으로 역할 구성원을 통해 pown(r_2)또는 비상 상황 시 할당된 bpown(r_1)을 통해 권한을 소유 할 수 있다. 정상 권한 소유권과 비상 상황 권한 소유권을 분리하려면 두 매핑을 통해 특정 사용자에 같은 권한이 할당되지 않도록 해야 한다.

   ∀ oprms ∈ OPRMS, r_1, r_2∈R with oprms ∈ bpown(r_1) 및 oprms ∈ pown(r_2) : $rown^{-1}(r_1) \cap rown^{-1}(r_2) = \varnothing$

정의 6. 비상 상황 동적 정확성

권한 기반 ERBAC 모델은 다음 요구 사항이 충족되는 경우 동적으로 정확하다고 한다.

1. (BTG-SSD 관계) 각 사용자는 정상적으로 역할 구성원을 통해 pown(r_2)또는 비상 상황 시 역할별 할당 bpown(r_1)을 통해 권한을 소유 할 수 있다. 일반 권한과 비상 상황 권한이 BTG-SSD 관계에 있다면, 일반 권한 소유권과 비상 상황 권한 소유권 매핑을 통해 특정 사용자에 BTG_SSD 관계에 있는 권한이 할당되지 않도록 해야 한다.

   ∀ oprms1, oprms2 ∈ OPRMS, oprms1 ∈ btg_ssd(oprms2) , r_1, r_2∈ R
   oprms1 ∈ bpown(r_1), oprms2 ∈ pown(r_2), $rown^{-1}(r_1) \cap rown^{-1}(r_2) = \varnothing$

2. (BTG-DSD 관계) 각 사용자는 정상적으로 역할 구성원을 통해 pown(r_2)또는 비상 상황 시 역할별 할당 bpown(r_1)을 통해 권한을 소유 할 수 있다. 일반 권한과 비상 상황 권한이 BTG-DSD 관계에 있다면, 일반 권한 소유권 매핑을 통해 권한을 가지고 비상 상황 권한 소유권 매핑을 통해 권한을 가지고 있는 특정 사용자가 BTG-DSD 관계에 있는 권한들을 동시에 활성화해서는 안 된다.

   ∀ oprms1, oprms2 ∈ OPRMS, oprms1 ∈ btg_dsd(oprms2), r_1, r_2∈ R
   oprms1 ∈ bpown(r_1), oprms2 ∈ pown(r_2),
   active_session^{-1} (r_1) ∩ active_session^{-1} (r_2) =∅

3. (BTG-Binding 관계) 각 사용자는 정상적으로 역할 구성원을 통해 pown(r_2)또는 비상 상황 시 역할 별 할당 bpown(r_1)을 통해 권한을 소유 할 수 있다. 일반 권한과 BTG 권한이 BTG-Binding 관계에 있다면, 일반 권한 소유권과 BTG 권한 소유권 매핑을 통해 특정 사용자에 BTG-Binding 관계에 있는 권한이 할당되어야 한다.

$\forall$ oprms1, oprms2 $\in$ OPRMS, oprms1 $\in$ btg_binding(oprms2) , r_1, $r_2 \in$ R
oprms1 $\in$ bpown(r_1), oprms2 $\in$ pown(r_2),
$rown^{-1}(r_1) = rown^{-1}(r_2)$

3.8 비상 상황 권한 기반 ERBAC 알고리즘

비상 상황에서의 권한 기반 ERBAC 알고리즘은 다음 Algorithm과 같다.

비상 상황이 발생하면 사용자가 시스템에 비상 상황 해결을 위하여 필요한 권한을 요청하고, 시스템은 이 순간부터 모든 작업을 기록한다. 시스템은 권한과 관련된 의무를 수행할 수 있는지, 없는지를 확인한 후 의무를 수행할 수 있으면 제어 모드로 전환하고 의무를 수행할 수 없는 상태이면 비 제어 모드로 전환한다.

그 후 사용자가 권한을 요청하면 사용자 신뢰 수준과 제한된 BTG 자원인지 여부를 점검한다. 사용자의 신뢰도를 검사하여 신뢰도가 'H'이고 요청한 권한이 제한된 BTG 자원이 아니면, BTG-SSD, BTG-DSD 제약 조건을 확인하여 위배되지 않는다면 BTG-Binding 제약조건을 확인한다. 사용자가 요청한 권한이 BTG-Binding 제약조건에 속하지 않으면 사용자에게 권한을 부여하고, BTG-Binding 제약조건에 속하면 요청한 권한이 속한 BTG-Binding의 권한들을 사용자에게 할당한다. 사용자는 이후에 비상 상황에서 할당받은 BTG 권한을 사용하여 비상 상황을 해결하게 된다. 그 후 사용자는 비상 상황이 해결됐다고 관리자에게 알리면 시스템의 상태가 제어 모드인 경우에는 로그 기록은 감사 기록에 자동으로 저장이 되고, 비 제어 모드인 경우는 로그 기록을 관리자가 수동으로 감사기록에 저장한 후에 시스템의 상태는 정상 상태로 복귀한다.

Algorithm : 비상 상황에서 권한 기반 ERBAC 알고리즘

```
1. User sets system status to emergency
2. From this moment on, all operation will be recorded.
3. Obligations related to emergency should be performed such as notifying the
   responsible manager and writing to the audit.
4. If (system is unable to fulfill its obligations) then
5.    Set the status of the system to uncontrolled Mode
6. else
7.    Set the status of the system to controlled Mode
8. End if
9. begin
10. User 'u' requests permission 'p'
11. If ((trust_level(u) = H) && ('p' ∉ restricted BTG resource)then
12.   begin
13.   If ((user role 'u' ∪ role of permission 'p')∉(BTG-SSD or BTG-DSD))
        If ((user role 'u' ∪ role of permission 'p') ∈ BTG-Binding) then
          begin
            Assign BTG-Binding'permission to role 'u'
        else
          begin
14.         Assign permission 'p' to role 'u'
15.       End if
16.   End if
17. End if
18. If (system status == controlled Mode) then
19.   Automatically save all activities to audit
20. else
21.   Manually save all activities to audit
22. End if
23. Remove permission of role
24. Rollback to normal situation
```

Ⅳ. 시나리오

4.1 의료 시나리오

이 절에서는 본 논문에서 제안된 모델을 다음과 같은 예시로 설명한다. 그림 2와 그림 3은 예시를 위한 역할 계층과 관리자 역할 계층을 보여준다.

표 1에서 총 관리자(A1)은 인턴(OPO)부터 병원장(D)까지의 역할을 관리하고 일반 의료 관리자(A2)는 전문의(OP2)부터 교수(OP3)까지의 역할을 관리한다. 또한, 정신과 의료 관리자(A3)는 정신과 전문의(PP2)부터 정신과 교수(PP3)까지의 역할을 관리한다. 그리고 VIP 의료 관리자(A4)는 VIP 전문의(PP2)부터 VIP 교수(PP3)까지의 역할을 관리한다.

또한, 특수체질 관리자(A5)는 특수 체질 전문의(SP2)부터 특수 체질 교수(SP3)까지의 역할을 관리한다. 마지막으로 일반 의료 관리자(A6)는 인턴(OP0)부터 레지던트(OP1)까지의 역할을 관리한다.

표 1. 관리 역할과 관리 범위
Table 1. Administrator role and range

Administration role	Administration range
A1	[OP0, D]
A2	[OP2, OP3]
A3	[PP2, PP3]
A4	[VP2, VP3]
A5	[SP2, SP3]
A6	[OP0, OP1]

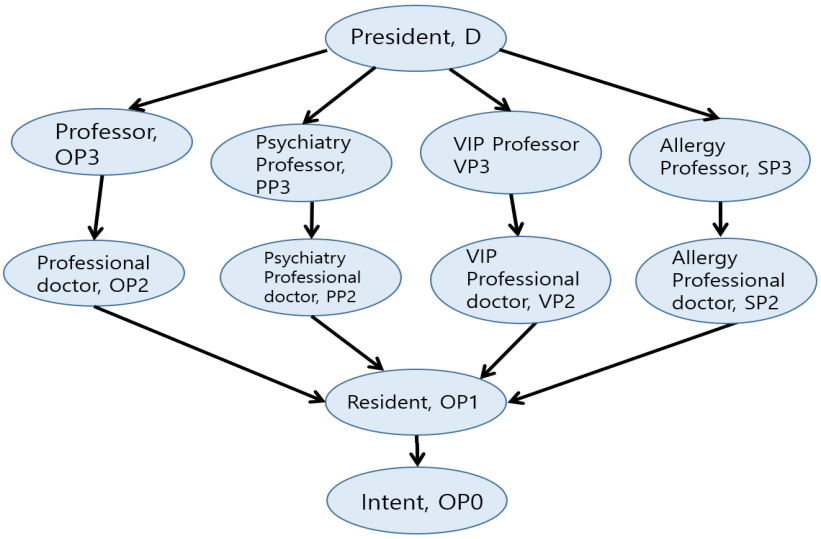

그림 2. 역할 계층
Fig. 2. Role hierarchy

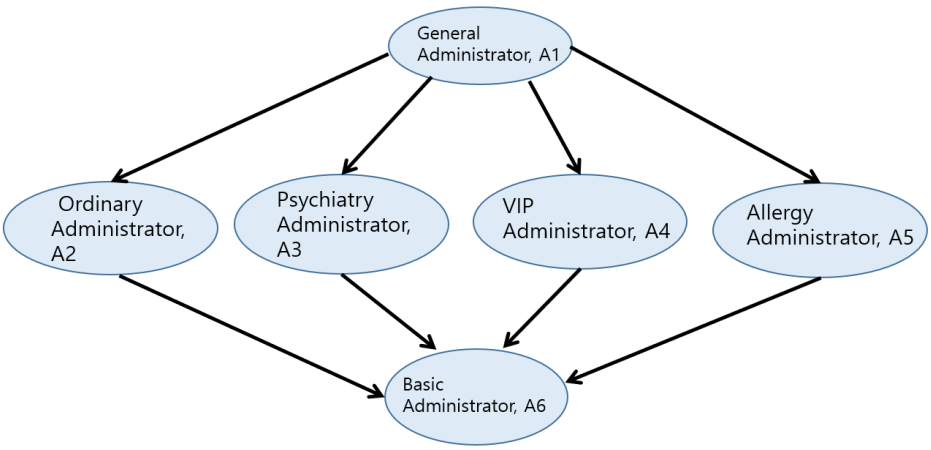

그림 3. 관리자 역할 계층
Fig. 3. Administrator role hierarchy

사용자 신뢰도 및 사용자-역할 할당은 표 1, 표 2와 같고, 역할-권한 할당은 표 3과 같다.

표 2. 사용자 신뢰도 및 사용자-역할 할당
Table 2. User trust-level and user-role allocation

User	Trust-level	Role
U0	H	M
U1	H	VP3
U2	H	PP3
U3	H	OP3
U4	H	VP2
U5	H	PP2
U6	H	OP2
U7	L	OP1
U8	L	OP0
U9	H	SP3
U10	H	SP2

표 3. 역할-권한 할당
Table 3. Role-permission allocation

Role	Permission
M	P0-VIP psychiatry confidential record
VP3	P1-VIP confidential record
PP3	P2-Psychiatry confidential record
OP3	P3-Confidential record
VP2	P4-VIP health record
PP2	P5-Psychiatry health record
OP2	P6-Health record
OP1	P7-Record
OP0	P8-Basic record
	P9-VIP allergy record write
SP3	P10-Psychiatry allergy record write
	P11-Allergy record write
	P12-VIP allergy record read
SP2	P13-Psychiatry allergy record read
	P14-Allergy record read

표 4. 권한 기반 SSD와 DSD 및 바인딩
Table 4. Permission-based SSD, DSD and binding

Permission	Permission -based SSD	Permission -based DSD	Permission -based binding
P1	P2	P3	P9
P2	P1, P3		P10
P3	P2	P1	P11
P4	P5	P6	P12
P5	P4. P6		P13
P6	P5	P4	P14

권한 기반 SSD와 DSD 및 바인딩의 예는 표 4와 같고, BTG-SSD와 BTG-DSD 및 BTG-바인딩의 예는 표 5과 같다. 표 6은 자원 접근 권한 및 제한된 BTG 자원 설정을 나타낸다.

표 5. BTG-SSD와 BTG-DSD 및 BTG-Binding
Table 5. BTG-SSD, BTG-DSD and BTG-binding

Permission	BTG_SSD	BTG_DSD	BTG_binding
P1	P2	P3	P9
P2	P1, P3		P10
P3	P2	P1	
P5			P14

표 6. 자원 접근 권한 및 제한된 BTG 자원 설정
Table 6. Access permission of resource and restricted BTG resource

Resource	Permission	Restricted BTG resource
Patient record	P3, P6, P7, P8	N
Psychiatry patient record	P2, P5, P7, P8	N
VIP patient record	P1, P4, P7, P8	N
Allergy patient record	P9, P10, P11, P12, P13, P14	N
VIP psychiatry confidential record	P0	Y

예시 1. 비상 상황 처리 예

VIP 환자가 응급치료가 필요한 상황이지만 병원에 VIP 환자를 전담하는 VIP 전문의(U4)가 없는 상황이고 일반 전문의(U6)만 있다고 가정한다. 하지만 사용자 U6는 표 2와 표 3에서 보인 바와 같이 일반 전문의로서 VIP 환자의 진료 기록에 대한 접근 권한을 가지고 있지 않다. 이 상황을 해결하기 위하여 일반 전문의(U6)는 시스템에 비상 상황이라고 알리고 VIP 환자의 의료 기록에 대한 접근을 위하여 P4 권한을 요청한다. 이 요청을 받은 시스템은 비상 상황에서의 의무 수행 여부에 따라 시스템의 상태를 제어 모드와 비 제어 모드로 설정한다. 시스템은 비상 상황에서 요청받은 P4 권한의 허가 여부를 결정하기 위하여 본 논문에서 제안한 권한 기반 ERBAC 알고리즘을 적용한다. 시스템은 U6의 비상 상황 P4 권한의 허가 여부를 결정하기 위하여 U6의 사용자 신뢰 수준을 점검하고, P4 권한이 제한된 BTG 자원에 속하지 않는지를 점검한다. 그 이후에 BTG-SSD 및 BTG-DSD 조건을 확인한 후 BTG

-Binding의 제약 조건을 확인한다. 현재 U6의 신뢰 수준이 'H'이고, P4가 제한된 BTG 자원에 속하지 않으며, U6가 역할 OP2를 통하여 가지고 있는 권한 P6와 BTG 요청 권한 P4가 표 6의 BTG-SSD 및 BTG-DSD조건에 위배되지 않고, BTG-Binding의 조건에 해당하지 않기 때문에, OP2 역할을 관리하는 관리자 역할 A2는 can_assignp를 수행하여 OP2에 P4를 할당한다. 사용자 U6는 인가받은 P4 권한을 사용하여 비상 상황을 해결하게 된다. 그 후 사용자는 비상 상황이 해결됐다고 관리자에게 알리면 관리자의 can_revokep를 수행하여 사용자에게 비상 상황 시 허가된 권한을 취소한다. 그 이후에 시스템의 상태가 제어 모드인 경우에는 로그 기록은 감사에 자동으로 저장이 되고, 비 제어 모드인 경우는 로그 기록을 관리자가 수동으로 감사에 저장 하고 시스템의 상태는 정상 상태로 복귀한다.

예시 2. BTG-SSD 적용 예

일반 환자가 사고로 인하여 환자의 기밀문서에 기초한 응급치료가 필요한 상황이지만 병원에 일반 환자를 치료할 수 있는 교수(U3) 및 전문의(U6)가 없는 상황이고 정신과 교수(U2)만 있다고 가정한다. 하지만 사용자 U2는 표 2와 표 3에서 보인 바와 같이 정신과 교수로서 일반 환자의 기밀문서 진료기록에 대한 접근 권한을 가지고 있지 않다. 이 상황을 해결하기 위하여 정신과 교수(U2)는 시스템에 비상 상황이라고 알리고 일반 환자의 기밀문서 에 대한 접근을 위하여 P3 권한을 요청한다. 이 요청을 받은 시스템은 비상 상황에서의 의무 수행 여부에 따라 시스템의 상태를 제어 모드와 비 제어 모드로 설정한다. 시스템은 비상 상황에서 요청받은 P3 권한의 허가 여부를 결정하기 위하여 본 논문에서 제안한 권한 기반 ERBAC 알고리즘을 적용한다. 정상 상황에서 사용자 U2는 역할 P3을 통하여 권한 P2를 할당받았기 때문에 비상 상황시 요청한 P3 권한은 BTG-SSD 제약조건(P2, P3)에 포함되기 때문에 할당될 수 없다. 그래서 시스템은 사용자 U2에게 접근 권한 P2 불가를 알리고 시스템의 상태는 정상 상태로 복귀한다.

예시 3. BTG-Binding 적용 예

사용자 일반 의료 전문의(U6)는 정신과 전문의(U5)가 없는 상황에서 정신과 환자를 치료해야 하는 비상 상황을 맞이하게 된다고 가정하자. 사용자 U6는 비상 상황을 처리하기 위하여 권한 P5를 요청한다고 가정하자. 비상 상황을 해결하기 위하여 U6는 시스템에 비상 상황이라고 알리고 정신과 환자의 의료 기록에 대한 접근을 위하여 P5 권한을 요청한다. 이 요청을 받은 시스템은 비상 상황에서의 의무 수행 여부에 따라 시스템의 상태를 제어 모드와 비 제어 모드로 설정한다. 시스템은 비상 상황에서 요청받은 P5 권한의 허가 여부를 결정하기 위하여 본 논문에서 제안한 권한 기반 ERBAC 알고리즘을 적용한다.

시스템은 U6의 비상 상황 P5 권한의 허가 여부를 결정하기 위하여 U6의 사용자 신뢰 수준을 점검하고, P5 권한이 제한된 BTG 자원에 속하지 않는지를 점검한다. 그 이후에 BTG -SSD 및 BTG-DSD 조건을 확인한 후 BTG-Binding의 제약 조건을 확인한다. 현재 U6의 신뢰 수준이 'H'이고, 권한 P5가 제한된 BTG 자원에 속하지 않기 때문에 BTG-SSD 및 BTG-DSD 조건을 확인한다. U6가 역할 OP2를 통하여 가지고 있는 권한 P6와 비상 상황 시 요청한 권한 P5가 일반 상황에서는 권한 기반 정적 의무 분리 제약조건(P5, P6)에 속하기 때문에 요청한 P6 권한이 허가되지 않지만, 비상 상황에서는 표 6의 BTG-SSD 및 BTG-DSD조건에 위배되지 않기 때문에 P5가 허가된다.

그 후에 비상 상황 시 요청한 P5 권한이 BTG-Binding(P5, P14)의 조건에 해당하기 때문에, 요청한 P5와 바인딩 된 P14권한까지 같이 수행되어야 하기 때문에 P5와 바인딩 된 P14를 같이 할당하게 된다. OP2 역할을 관리하는 관리자 역할 A2는 can_assignp를 수행하여 OP2에 P5와 P14를 할당한다. 사용자 U6는 인가받은 P5와 P14 권한을 사용하여 비상 상황을 해결하게 된다. 그 후 사용자는 비상 상황이 해결됐다고 관리자에게 알리면 관리자 A2는 can_revokep를 수행하여 사용자에게 비상 상황 시 허가된 권한을 취소한다. 그 이후에 시스템의 상태가 제어 모드인 경우에는 로그 기록은 감사에 자동으로 저장이 되고, 비 제어 모드인 경우는 로그 기록을 관리자가 수동으로 감사에 저장하고 시스템의 상태는 정상 상태로 복귀한다.

Ⅴ. 비교 평가

표 7은 접근 제어 모델 비교를 나타낸다. 비상 상황 시를 고려한 접근제어 모델들로 기존의 연구들과 본 논문에서 제안한 권한 기반 ERBAC 모델을 비교한다.

표 7에서 보이는 바와 같이 본 논문에서 제안한 권한 기반 ERBAC 모델은 비상 상황 시에 사용자가 비상 상황 권한 요청 시 해당 기관의 특성을 고려한 사용자 신뢰 수준을 고려하고, 비상 상황에서만 적용되는 권한 기반의 BTG-SSD, BTG-DSD 및 BTG-Binding의 개념을 새로 도입하여 비상 상황 시에도 상세한 접근 제어가 가능하도록 한다. 그리고 비상 상황 시에 권한을 기반으로 필요한 권한만을 관리자가 허가하도록 처리함으로써 최소 권한 원칙을 준수하고 관리자의 부담을 줄이도록 한다. 또한 비상 상황 시 행하여야 하는 의무 수행 여부에 따라서 시스템의 상태를 제어 모드, 비 제어 모드 두 가지의 경우로 나누어 시스템이 의무를 수행할 수 있는 상황이면 제어 모드로 비상 상황에서의 작업 기록이 모두 자동으로 감사기록을 수행하도록 하고, 의무를 수행하지 못하는 경우에는 비 제어 모드로 비상 상황이 종료된 후 관리자가 직접 감사 기록을 수행하도록 하여 의무를 수행하지 못하는 경우에도 비상 상황 처리가 가능하도록 하게 한다. 추가로 제한된 BTG 자원 개념을 도입하여, 비상 상황에서도 특정 자원에는 절대 접근을 할 수 없도록 하여 시스템의 안전성으로 높이도록 한다. 이와 반면에 기존의 연구들은 표 7에서 보이는 바와 같이 각기 다른 단점들을 나타내고 있다.

Ⅵ. 결 론

본 논문에서는 비상 상황 시에 사용자의 권한 요청에 대한 접근 제어를 수행하기 위하여 새로운 권한 기반의 ERBAC 모델을 제안하였다. 제안된 권한을 기반으로 한 ERBAC 모델은 비상 상황 시 사용자가 필요로 하는 권한을 요청하는 경우 필요 권한만 현재 사용자의 활성화된 역할에 할당하여 최소 권한 원칙을 지킨다.

또한, 비상 상황을 하나의 메커니즘으로 일원화하고 사용자가 요청하는 특정 권한만 부여하고 제거함으로써 관리모델의 부담을 줄였다. 또한 비상 상황 시에도 세밀한 접근제어를 수행하기 위하여 비상 상황에서만 적용되는 권한 기반의 BTG-SSD, BTG-DSD 및 BTG-Binding의 개념을 새로 도입하여 비상 상황 시에도 상세한 접근 제어가 가능하도록 하였다. 그리고 사용자가 비상 상황 권한 요청 시 해당 기관의 특성을 고려한 사용자 신뢰 수준을 고려하고 접근하는 자원에 제한을 두어 비상 상황이어도 접근을 할 수 없도록 설정하여 사용자의 오용 및 남용을 방지하도록 하여 시스템의 안전성을 높였다. 향후에는 하나의 기관이 아닌 서로 다른 기관의 사용자들이 비상 상황 시 다른 기관의 권한을 사용할 수 있는 연구를 수행할 예정이다.

References

[1] Ravi S. Sandhu, Edward J. Coynek, Hal L. Feinsteink, and Charles E. Youman, "Role-Based Access Control Models", IEEE Computer, Vol. 29, No. 2, pp. 38-47 Feb. 1996.

표 7. 접근 제어 모델 비교
Table 7. Comparison of access control models

Features	BTG-RBAC [3]	Process RBAC [4]	E-RBAC [5]	Exc-RBAC [6]	Controlled BTG [9]]	Proposed model
BTG least privileges	X	V	X	X	V	V
BTG administrator load	V	X	V	V	X	V
BTG facility policies	X	X	V	V	X	V
BTG uncontrolled mode	X	X	X	X	V	V
BTG-SSD	X	V	X	X	X	V
BTG-BINDING	X	V	X	X	X	V
Restricted BTG resource	X	X	X	X	V	V

[2] Ravi Sandhu, David Ferraioloy, and Richard Kuhny, "The NIST Model for Role-Based Access Control: Towards A Unified Standard", Fifth ACM Workshop on Role-Based Access Control, pp. 47-63, Jul. 2000.

[3] Ana Ferreira, David Chadwick, Gansen Zao, Pedro Farinha, Ricardo Correia, and Rui Chilro, "How to securely break into RBAC: the BTG-RBAC model", 2009 Annual Computer Security Applications Conference, pp. 23-31, Dec. 2009.

[4] Sigrid Schefer-Wenzl, and Mark Strembeck, "Generic Support for RBAC Break-Glass Policies in Process-Aware Information Systems", Proceedings of the 28th Annual ACM symposium on Applied Computing, Coimbra Portugal, pp. 1441-1446, Mar. 2013. https://doi.org/10.1145/2480362.2480631.

[5] Fatemeh Nazerian, Homayun Motameni, and Hossein Nematzadeh, "Emergency role-based access control (E-RBAC) and analysis of model specifications with alloy", Journal of Information Security and Applications, Vol. 45, pp. 131-142, Apr. 2019. https://doi.org/10.1016/j.jisa.2019.01.008.

[6] Abdelkrim, BOUADJEMI, Mustapha Kamel, and ABDI, "Towards an Extension of RBAC Model", International Journal of Computing and Digital Systems, Vol. 10, pp. 1-11, Jul. 2020.

[7]. Muhammad Asif Habib, Nasir Mahmood, Muhammad Shahid, Muhammad Umar Aftab, Uzair Ahmad, and C. Muhammad Nadeem Faisal, "Permission Based Implementation of Dynamic Separation of Duty (DSD) in Role Based Access Control (RBAC)", In Proceedings of the 8th International Conference on Signal Processing and Communication Systems, Gold Coast, QLD, Australia, pp. 1-10, Dec. 2014. https://doi.org/10.1109/ICSPCS.2014.7021054.

[8] Muhammad Umar Aftab, Zhiguang Qin, Negalign Wake Hundera, Oluwasanmi Ariyo, Zakria, Ngo Tung Son, and Tran Van Dinh, "Permission-Based Separation of Duty in Dynamic Role-Based Access Control Model", Symmetry, Vol. 11, No. 5, p. 669, May 2019. https://doi.org/10.3390/sym11050669.

[9] Qais Tasali, Christine Sublett, and Eugene Y. Vasserman, "Controlled BTG: Toward Flexible Emergency Override in Interoperable Medical Systems", EAI Endorsed Transactions on Security and Safety, Vol. 6, No. 22, pp. 3-17, Feb. 2020. https://doi.org/10.4108/eai.13-7-2018.163213.

[10] Baoping Wang, Guang Zhao, Jun Liu, and Xingang Zhang, "Available Separation-of-Duty Policies in Access Control", 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing, Wuhan, China, pp. 290-293, Apr. 2010. https://doi.org/10.1109/NSWCTC.2010.73.

[11] RAVI SANDHU, VENKATA BHAMIDIPATI, and QAMAR MUNAWER, "The ARBAC97 Model for Role-Based Administration of Roles", ACM Transactions on Information and System Security, Vol. 2, No. 1, pp. 105-135, Feb. 1999. https://doi.org/10.1145/300830.300839.

[12] S. Schefer-Wenzl, H. Bukvova, and M. Strembeck, "A review of delegation and break-glass models for flexible access control management", International conference on Business Information Systems, Larnaca, Cyprus, pp. 93-104, May 2014.

[13] Yu-Dong Hwang and Dong-Gue Park, "Extended GTRBAC Model for Access Control of Integrated Video Surveillance System", Journal of KIIT, Vol. 15, No. 7, pp. 91-100, Jul. 2017. https://doi.org/10.14801/jkiit.2017.15.7.91.

저자소개

백 경 동 (Kyeong-Dong Baek)

2013년 3월 ~ 2020년 2월 :
순천향대학교
정보통신공학과(학부)
2020년 3월 ~ 현재 : 순천향대학교
정보통신공학과(석사과정)
관심분야 : 제어 시스템, 모바일
애플리케이션, 시스템 보안

박 동 규 (Dong-Gue Park)

1992년 2월 : 한양대학교
전자공학과(공학박사)
1992년 3월 ~ 현재 : 순천향대학교
정보통신공학과 교수
관심분야 : 제어시스템 보안, 네트
워크보안, 시스템 보안, 모바일
보안